Kentucky Geographic Tools ActiveX Component
Version 3.1

Component: kyGeoTools
Object: kyGeoPosition

Methods and Properties:

Name	Status	DataType	Description
SetPath	Method	String Input	Sets the path where the `kyGeoTools.bin` file resides on the local system. Requires a string parameter defining the path to the bin file. Usage: SetPath(*ByVal PathSpec As String*).
WestPositive	Read/Write	Boolean	Specifies if west longitude values are to be returned as a positive (True) or negative (False) value (default = True).
x	Read/Write	Double	Gets or Sets the x *(Easting or Longitude)* coordinate of the position object (default = 0).
y	Read/Write	Double	Gets or Sets the y *(Northing or Latitude)* coordinate of the position object (default = 0).
Carter	Properties Container		Container for Carter Coordinates properties (see page 3).
Datum	Read/Write	Enum	Gets or Sets the current object's datum as a **kyGeoDatum** enumeration value as follows: `kyNAD83 = 0` NAD83 (default). `kyNAD27 = 1` NAD27 **Note:** The object's Datum and Units will be automatically changed based on the Projection property setting.
Projection	Read/Write	Enum	Gets or Sets the current object's projection as a **kyGeoProjection**. enumeration value as follows: `kySingleZone = 0` Kentucky Single Zone. Datum is subsequently forced to `kyNAD83`. `kyNorthZone = 1` Kentucky North Zone `kySouthZone = 2` Kentucky South Zone `kyUTM16 = 3` UTM Zone 16 `kyUTM17 = 4` UTM Zone 17 `kyGeographic = 5` Geographic (default). Units are subsequently forced to `kyGeographic`. `kyCarter = 6` Carter Coordinate System. Units are subsequently forced to `kyGeographic`. `moEastZone = 7` Missouri East Zone `ilWestZone = 8` Illinois West Zone `ilEastZone = 9` Illinois East Zone `inWestZone = 10` Indiana West Zone `inEastZone = 11` Indiana East Zone `ohSouthZone = 12` Ohio South Zone `wvSouthZone = 13` West Virginia South Zone `vaSouthZone = 14` Virginia South Zone `tnSingleZone = 15` Tennessee Single Zone

Properties (continued):

Name	Status	DataType	Description
Units	Read/Write	Enum	Gets or Sets the current object's mapping units as a **kyGeoUnits** enumeration value as follows: `kyDecimalDegrees = 0` Decimal Degree (default) `kyUSFt = 1` U.S. Survey Feet `kyMeters = 2` Meters `kyIntlFt = 3` International Feet `kyMiles = 4` Miles `kyKilometers = 5` Kilometers  **Note:** The decimal degree unit is valid only for the `kyGeographic` and `kyCarter` **Projection** options.
ScaleFactor	Read Only	Double	Returns the object's scale factor for the current projection. A scale factor of one (1) is returned for `kyGeographic`.
Convergence	Read Only	Double	Returns the object's convergence angle in decimal degrees for the specified projection. A convergence of zero (0) is returned for the `kyGeographic`.

InBounds	Read Only	Boolean	Returns True if the position lies within the following NAD27 envelope:
			
			`From 36.0 deg to 39.5 deg North Latitude`
			`From 81.5 deg to 90.0 deg West Longitude`

StateCount	Read Only	Long	Returns the number of states the position lies within, including on boundary lines. Returns 0 if the position is out of bounds. Maximum = 3
StateName	Read Only	String Array	Array (0 to 2)* of state names. Index 0 is Kentucky for positions lying within or on the Kentucky state boundary.
StateFIPS	Read Only	Long Array	Array (0 to 2)* of state FIPS codes. Index 0 is 21 (Kentucky) for positions lying within or on the Kentucky state boundary.
CountyCount	Read Only	Long	Returns the number of counties the position lies within, including on boundary lines. Returns 0 if the position lies outside Kentucky. Maximum = 3
CountyName	Read Only	String Array	Array (0 to 2)* of county names.
CountyFIPS	Read Only	Long Array	Array (0 to 2)* of county FIPS codes.
QuadCount	Read Only	Long	Returns the number of USGS 7.5 Minute Quadrandles the position lies within, including on boundary lines. Returns 0 if the position lies outside the quadrangle coverage area for Kentucky. Maximum = 4
QuadTile	Read Only	String Array	Array (0 to 3)* of quadrangle tile designations.
QuadName	Read Only	String Array	Array (0 to 3)* of quadrangle names.
QuadState	Read Only	String Array	Array (0 to 3)* of state abbreviations.

* The default index for all property arrays is 0, thus *obj*.Property is the same as *obj*.Property(0)

Notes on SetPath: Each time a new instance of a **kyGeoPosition** object is created **kyGeoTools** attempts to find the `kyGeoTools.bin` data file by searching the current folder, the default Windows and System folder, and the folders defined within the Path environment variable. If that attempt is not successful then you must specify the path where the data file resides through this property. Otherwise, the component will return an error message each time it encounters a property that requires data in the Bin file. Calling the **SetPath** method is not required if the `kyGeoTools.bin` file is stored in any one of the above listed folders, otherwise, you need only successfully call this method one time per instance of the object.

Enumerated Constants:

Type kyGeoDatum

```
kyNAD83  =  0
kyNAD27  =  1
```

Type kyGeoProjection

```
kySingleZone  =  0
kyNorthZone   =  1
kySouthZone   =  2
kyUTM16       =  3
kyUTM17       =  4
kyGeographic  =  5
kyCarter      =  6
moEastZone    =  7
ilWestZone    =  8
ilEastZone    =  9
inWestZone    =  10
inEastZone    =  11
ohSouthZone   =  12
wvSouthZone   =  13
vaSouthZone   =  14
tnSingleZone  =  15
```

Type kyGeoUnits

```
kyDecimalDegrees  =  0
kyUSFt            =  1
kyMeters          =  2
kyIntlFt          =  3
kyMiles           =  4
kyKilometers      =  5
```

Properties Container: Carter

The **Carter** container provides Read/Write access to the following Carter Coordinates properties:

Name	Status	DataType	Description
BlockLetter	Read/Write	String	Gets or Sets the letter designation for the 5-minute block. The valid range for setting this value is A to Z, AA to GG (upper or lower case). This property always returns an upper case value. Attempting to set this property with values outside the valid range will result in an error exception condition.
BlockNumber	Read/Write	Long	Gets or Sets the number designation for the 5-minute block. The valid range for setting this value is 0 to 92. Attempting to set this property with values outside the valid range will result in an error exception condition.
CellNumber	Read/Write	Long	Gets or Sets the number designation for the 1-minute cell within the associated Block. The valid range for setting this value is 1 to 25. Attempting to set this property with values outside the valid range will result in an error exception condition.
FNL	Read/Write	Double	Gets or Sets the offset distance in feet to be measured south from the north line of the applicable cell.
FSL	Read/Write	Double	Gets or Sets the offset distance in feet to be measured north from the south line of the applicable cell.
FEL	Read/Write	Double	Gets or Sets the offset distance in feet to be measured west from the east line of the applicable cell.
FWL	Read/Write	Double	Gets or Sets the offset distance in feet to be measured east from the west line of the applicable cell.

LocationString	Read Only	String	Returns the location string for the Carter Coordinates defined by the current Carter properties. This string is formatted as follows:
			
			`Block-Cell dist-FNL/FSL dist-FEL/FWL`
			
			`Example: G34-02 123-FSL 456-FWL`
			
			Notes: Offset distances are always given in feet.
			
			 
			See Page 5 for further details pertaining to behavior of Cater Coordinate values.
			
			`Out of Bounds` is returned for positions that fall outside the following geographic envelope:
			
			`From 36° 30′ to 39° 15′ North Latitude`
			`From 81° 55′ to 89° 35′ West Longitude`

FAQ on Carter Coordinate System

Method: Convert

Converts the datum, projection, and/or units of the given **kyGeoPosition** object and returns a **Long** specifying an error code. Conversion includes computing new *x* and *y* coordinate values for the object and re-defining its datum, projection, and/or units properties. Conversion includes NADCON transformations for datum changes.

Usage: *object*.**Convert** [*toDatum*], [*toProjection*], [*toUnits*]

Alternate Usage: *RCode* = *object*.**Convert**([*toDatum*], [*toProjection*], [*toUnits*])

The **Convert** method syntax has these arguments:

Part	Description
RCode	The return code returned by the method as a **Long**. Possible values are: 0 = Successful conversion, no errors encountered. 4 = Position is out of bounds. The current position is not changed. This code is returned when a NADCON operation (forward or reverse) is attempted on a position that falls outside the following envelope: NAD27 North Latitude: 36° 00' to 39° 30' NAD27 West Longitude: 81° 30' to 90° 00' Note: In previous versions error codes of 1, 2, and 3 were returned for *InvalidToDatum*, *InvalidToProjection*, and *InvalidToUnits* conditions respectively. An error exception is now invoked for those conditions.
object	A **kyGeoPosition** object.
toDatum	Optional. Specifies the new **kyGeoDatum** enumeration as outlined above. The default is no change.
toProjection	Optional. Specifies the new **kyGeoProjection** enumeration as outlined above. The default is no change.
toUnits	Optional, Specifies the new **kyGeoUnits** enumeration as outlined above. toUnits are forced to `kyDecimalDegrees` for ***toProjection*** = `kyGeographic` and `kyCarter`. See Note 4 on the next page for details on other scenarios.

Note: When ***toProjection*** is set to `kyCarter`, the Convert method, upon completion, will set the ***x*** and ***y*** properties of the parent **kyGeoPosition** object to the corresponding `kyGeographic` coordinates (ddLng and ddLat respectively) as defined in the applicable ***toDatum***, and units will be set to `kyDecimalDegrees`.

General Notes:

1. The SetPath method should not be needed if the install program places the kyGeoTools.bin file in the current Windows System32 folder. If a particular installation requires this file to reside in a different folder, it should only be used one time per new instance of the object and should not be used within a loop.

2. The Convert method will work with negative decimal degree longitude values and will preserve the algebraic sign of longitude for changes in datum if originally set by the user, however, it always returns a positive longitude value when converting a position from state plane or UTM to its geographic (ddLatLong) equivalent. You should be aware of this characteristic when converting positions and require or prefer negative west longitude values.

3. The only valid Datum for the kySingleZone projection is kyNAD83. Thus, if the kySingleZone projection is specified then the Datum property will automatically be set to kyNAD83 when a calculation is required (i.e. for ScaleFactor, Convergence, or Convert).

4. When converting from the geographic projection to state plane or UTM without specifying a change in units, new units will be assigned to the position as follows:

   - Units are set to US Survey Feet for state plane projections.
   - Units are set to Meters for UTM projections.

Notes on behavior of Carter Coordinate values:

When Carter Coordinate parameters are set, only the BlockLetter, BlockNumber, and CellNumber parameters are checked for validity. The FNL, FSL, FEL, FWL offsets are not validated for values that would result in positions that actually lie within the referenced cell. The conversion routines computes the ellipsoidal radius values for the latitudinal and longitudinal components based on the latitude of the specified cell's centroid to establish the commensurate distance per latitude or longitude value, then applies those results to the lower left coordinate of the specified cell. Thus, it is important, but not required, that offset values fall within or reasonably close the nominal range for a given cell. When geographic or projected positions are converted into Carter Coordinate values, offset values are based on the shortest distance from the applicable edges of the computed cell. In other words, if a position lies at or below the centroid of a cell, then the offset is computed from the south line. If the position falls above the centroid of the cell, then the offset is computed from the north line, and likewise with respect to the FEL/FWL offsets for positions lying west or east of the cell centroid. This will have the effect of some Carter Coordinate values changing during round-trip conversions because any position may have more than one valid Carter Coordinate designation.

Once a Carter position has been successfully established, the ***x*** and ***y*** components of the **kyGeoPosition** object are set to the corresponding `kyGeographic` values (ddLatLng) for the specified datum, and units are set to `kyDecimalDegrees`.

Component: kyGeoTools
Object: kyGeoPosition

Example 1: This example returns the location (state, county, quadrangle) of a position:

```
Dim mStr as String
Dim i as Long, j as Long
Dim thePosition as New kyGeoPosition
With thePosition
  Call .SetPath("c:\MyPreferredPath") ' Only if needed.
  .x = -84.659875        ' Default = NAD83 DD longitude
  .y = 36.895214         ' Default = NAD83 DD latitude
  If .InBounds Then
    For i = 0 To .StateCount - 1
      mStr = mStr & .StateName(i) & vbCrLf
      If .StateFIPS(i) = 21 Then
        For j = 0 To .CountyCount - 1
          mStr = mStr & "  " & .CountyName(j) & vbCrLf
        Next j
      End If
    Next i
    For i = 0 To .QuadCount - 1
      mStr = mStr & .QuadTile(i) & "  "
      mStr = mStr & .QuadName(i) & ", "
      mStr = mStr & .QuadState(i) & vbCrLf
    Next i
  Else
    mStr = "Out of Bounds"
  End If
End With
MsgBox mStr
```

Example 2: The following example opens an ASCII file containing a list of NAD27 Kentucky South Zone northing and easting values, converts them to NAD83 Kentucky Single Zone, and writes the converted values to a separate ASCII file.

```
Dim Northing As Double, Easting As Double
Dim thePosition As New KyGeoPosition
Open InputFile For Input As #1
Open OutputFile For Output As #2
  Do Until EOF(1)
    Input #1, Northing, Easting
    With thePosition
      .x = Easting
      .y = Northing
      .Projection = kySouthZone
      .Datum = kyNAD27
      .Units = kyUSFt
      If .InBounds Then
        .Convert(kyNAD83, kySingleZone, kyUSFt)
        Print #2, .y; ","; .x
      End If
    End With
  Loop
Close
```

Example 3: The following example begins with the Lat and Long values for NGS Control Monument 218, PID GZ0128 in Fayette County, Kentucky, then converts the position to the Kentucky Single Zone projection.

The NGS published values for this monument are as follows:

```
GZ0128* NAD 83(1993)-  37 54 23.56139(N)    084 21 23.32041(W)     ADJUSTED
GZ0128;                    North         East     Units   Scale      Converg.
GZ0128;SPC KY1Z     - 3,856,664.47  5,323,268.40   sFT  0.99990508 +0 51 20.0
GZ0128;SPC KY N     -   148,065.78  1,609,694.42   sFT  1.00000964 -0 03 58.5
GZ0128;SPC KY S     - 2,216,189.15  2,042,471.67   sFT  0.99999522 +0 50 42.4
GZ0128;UTM  16      - 4,198,741.328   732,408.809   MT  1.00026531 +1 37 29.4
GZ0128
GZ0128                          SUPERSEDED SURVEY CONTROL
GZ0128
GZ0128  NAD 83(1986)-  37 54 23.56787(N)    084 21 23.32559(W) AD(       ) 2
GZ0128  NAD 27      -  37 54 23.28400(N)    084 21 23.56900(W) AD(       ) 2
```

```
'Note: The default values for new position objects are:
'      kyNAD83, kyGeographic, kyDecimalDegrees

  Dim thePosition As New KyGeoPosition
  With thePosition
    Call .SetPath("c:\MyPreferredPath") ' only if needed
    .x = 84# + 21# / 60# + 23.32041 / 3600#
    .y = 37# + 54# / 60# + 23.56139 / 3600#
    .Convert , kySingleZone, kyUSFt
    Debug.Print Format(.y, "#,###,###.00"); "  "; _
                Format(.x, "#,###,###.00"); "  "; _
                Format(.ScaleFactor, "#0.00000000"); "  "; _
                Format(.Convergence, "#0.00000000")
  End With
```

The above code returns `3,856,664.47  5,323,268.40  0.99990508  0.85556781`

Example 4: This example converts the position given in the above example to NAD83 UTM Zone 16 in meters, then returns the Carter Coordinate for that position:

```
  Dim thePosition As New KyGeoPosition
  With thePosition
    Call .SetPath("c:\MyPreferredPath") ' only if needed
    .x = 84# + 21# / 60# + 23.32041 / 3600#
    .y = 37# + 54# / 60# + 23.56139 / 3600#
    .Convert , kyUTM16, kyMeters
    Debug.Print Format(.y, "#,###,###.000"); "  "; _
                Format(.x, "#,###,###.000"); "  "; _
                Format(.ScaleFactor, "#0.00000000"); "  "; _
                Format(.Convergence, "#0.00000000")
    .Convert , kyCarter, kyUSFt
    Debug.Print .Carter.LocationString
  End With
```

The above code returns:

```
4,198,741.328  732,408.809  1.00026531  1.62484153
Q62-02 2383-FSL 1869-FEL
```